ARM Processor Cortex™-A17 MPCore

Product Revision r0

Software Developers Errata Notice

Non-Confidential - Released

ARM

Software Developers Errata Notice

Non-Confidential Proprietary Notice

Web Address

http://www.arm.com

Feedback on content

If you have any comments on content, then send an e-mail to errata@arm.com . Give:

- the document title
- the document number, ARM-EPM-061775
- the page numbers to which your comments apply
- a concise explanation of your comments.

ARM also welcomes general suggestions for additions and improvements.

Release Information

Errata are listed in this section if they are new to the document, or marked as "updated" if there has been any change to the erratum text in Chapter 2. Fixed errata are not shown as updated unless the erratum text has changed. The summary table in section 2.2 identifies errata that have been fixed in each product revision.

18 Mar 2014: Changes in Document v1

Page	Status	ID	Cat	Rare	Summary of Erratum
16	New	824719	CatC		Stage 2 fault might not be reported when address translation for a Cache Maintenance Operation uses a NoAccess Domain

22 May 2014: Changes in Document v2

Page	Status	ID	Cat	Rare	Summary of Erratum
7	New	826370	CatA		ICIMVA operation executed concurrently with a broadcast ICI or TLBI operation may cause either operation to be performed
11	New	826369	CatB	Rare	CPU might execute incorrect instructions following a TLBI+DSB sequence
12	New	827821	CatB	Rare	CPU might execute stale instructions following broadcast ICI operations
13	New	828419	CatB	Rare	CPU might execute an Instruction Cache Maintenance Operation by MVA with an incorrect NS descriptor
14	New	828420	CatB	Rare	In a multi-cluster system, Cache Clean by MVA to PoC might not be correctly executed

17 Jul 2014: Changes in Document v3

Page	Status	ID	Cat	Rare	Summary of Erratum
9	New	830075	CatB		Instructions might be fetched from unauthorized memory region when MMU is off
10	New	831171	CatB		A DSB might complete before the end of a broadcasted TLB or Icache invalidation
15	New	832071	CatB	Rare	Read after read issue with Streaming store and concomitant store by another CPU
17	New	831169	CatC		ECC error on L2 DATA RAMs might report an erroneous error index

Contents

Chapter 1.
Introduction

This chapter introduces the errata notice for the ARM Cortex-A17 MPCore processor.

1.1. Scope of this document

This document describes errata categorized by level of severity. Each description includes:

- the current status of the defect
- where the implementation deviates from the specification and the conditions under which erroneous behavior occurs
- the implications of the erratum with respect to typical applications
- the application and limitations of a 'work-around' where possible

This document describes errata that may impact anyone who is developing software that will run on implementations of this ARM product.

1.2. Categorization of errata

Errata recorded in this document are split into the following levels of severity:

Table 1 Categorization of errata

Errata Type	Definition
Category A	A critical error. No workaround is available or workarounds are impactful. The error is likely to be common for many systems and applications.
Category A(rare)	A critical error. No workaround is available or workarounds are impactful. The error is likely to be rare for most systems and applications. Rare is determined by analysis, verification and usage.
Category B	A significant error or a critical error with an acceptable workaround. The error is likely to be common for many systems and applications.
Category B(rare)	A significant error or a critical error with an acceptable workaround. The error is likely to be rare for most systems and applications. Rare is determined by analysis, verification and usage.
Category C	A minor error.

Chapter 2.
Errata Descriptions

2.1. Product Revision Status

The r*n*p*n* identifier indicates the revision status of the product described in this book, where:

r*n* Identifies the major revision of the product.

p*n* Identifies the minor revision or modification status of the product.

2.2. Revisions Affected

Table 2 below lists the product revisions affected by each erratum. A cell marked with **X** indicates that the erratum affects the revision shown at the top of that column.

This document includes errata that affect revision r0 only.

Refer to the reference material supplied with your product to identify the revision of the IP.

Table 2 Revisions Affected

ID	Cat	Rare	Summary of Erratum	r0p0
826370	CatA		ICIMVA operation executed concurrently with a broadcast ICI or TLBI operation may cause either operation to be performed	X
831171	CatB		A DSB might complete before the end of a broadcasted TLB or Icache invalidation	X
830075	CatB		Instructions might be fetched from unauthorized memory region when MMU is off	X
832071	CatB	Rare	Read after read issue with Streaming store and concomitant store by another CPU	X
828420	CatB	Rare	In a multi-cluster system, Cache Clean by MVA to PoC might not be correctly executed	X
828419	CatB	Rare	CPU might execute an Instruction Cache Maintenance Operation by MVA with an incorrect NS descriptor	X
827821	CatB	Rare	CPU might execute stale instructions following broadcast ICI operations	X
826369	CatB	Rare	CPU might execute incorrect instructions following a TLBI+DSB sequence	X
831169	CatC		ECC error on L2 DATA RAMs might report an erroneous error index	X
824719	CatC		Stage 2 fault might not be reported when address translation for a Cache Maintenance Operation uses a NoAccess Domain	X

2.3. Category A

826370: ICIMVA operation executed concurrently with a broadcast ICI or TLBI operation may cause either operation to be performed

Category A

Products Affected: Cortex-A17 MPCore.

Present in: r0p0

Description

ICIMVA operation executed concurrently with a broadcast ICI or TLBI operation may cause either operation to be performed using incorrect attributes

Configurations affected

This erratum affects all configurations of Cortex-A17 in systems containing more than 1 CPU (MP2/MP3/MP4 cluster or ACE system)

Conditions

The following conditions have to be met in order for the erratum to occur:

- The local ICIMVA operation needs to perform a Page Table Walk
- The result of the Page Table Walk for the ICIMVA operation is arbitrated in the same cycle as a broadcast ICI or TLBI operation
- Both the local ICIMVA operation and the broadcast operation are affected if the broadcast operation is either:
  - an ICIALL operation from a CPU in the same cluster
  - a Virtual Instruction Cache Invalidate received on ACE
  - a Physical Instruction Cache Invalidate received on ACE that does not specify a Virtual Index
- Only the local ICIMVA operation is affected if the broadcast operation is either:
  - an ICIMVA operation from a CPU in the same cluster
  - a Physical Instruction Cache Invalidate received on ACE that also specifies a Virtual Index

Implications

The affected operations will be performed using incorrect attributes, causing the expected operation not to be performed. As a result, instruction fetches may see stale instructions.

Workaround

No workaround.

There are no errata in this category

2.4. Category A (Rare)

There are no errata in this category

2.5. Category B

830075: Instructions might be fetched from unauthorized memory region when MMU is off

Category B

Products Affected: Cortex-A17 MPCore.

Present in: r0p0

Description

When disabling the MMU, architecture expects that instructions are only fetched from:

- Addresses already seen by previously executed branches.
- Addresses in the 4k region after the currently executing instruction.

Under specific conditions, a limited number of instructions might be fetched from the target of a branch executed before MMU was stopped, which is incorrect.

Configurations affected

This erratum affects all configurations of Cortex-A12 and Cortex-A17.

Conditions

The processor must be executing a code sequence having the following characteristics:

- A branch is executed, and put in the BTAC.
- The MMU is turned off for the translation regime used by the CPU. This can happen in the following case:
  - Changing the value of the system control register for this translation regime.
  - Changing the translation regime by changing the security state or privilege level.
- One of the four subsequent instructions has the same VA as a previously executed branch.

If these conditions are met, the CPU can issue one access to an unauthorized memory region.

Implications

The affected CPU might fetch instructions from a forbidden unauthorized memory region.

Workaround

A software workaround exists for the cases where the MMU is turned off by changing the system control register, or where an exception returns from a more privileged translation regime with its MMU enabled to another less privileged translation regime. It consists of performing a Branch Predictor Invalidate All operation before doing the ISB (in the case where the system control register is changed) of the exception return instruction (in the case of a change of the translation regime).

This workaround does not apply in the case where the less privileged translation regime has its MMU enabled while it is disabled for a more privileged translation regime.

831171: A DSB might complete before the end of a broadcasted TLB or Icache invalidation

Category B

Products Affected: Cortex-A17 MPCore.

Present in: r0p0

Description

Within rare timing constraints, a DSB following a TLB or Icache invalidation might complete before the TLB or ICache invalidation has effectively completed on other CPUs within the cluster.

Configurations affected

This erratum affects all configurations of Cortex-A17 in a system with three or more CPUs.

Conditions

The following conditions have to be met for the erratum to occur:

- CPU1 performs a TLB or ICache maintenance operation followed by a DSB.
- CPU2 performs a TLB or ICache maintenance operation followed by a DSB.
- CPU0 is running.

Implications

When executing the sequence of transactions described above and when particular timing conditions occur, CPU1 might complete the DSB before CPU0 has effectively performed the broadcasted TLB or Icache invalidation.

Workaround

- Issue the TLB or ICache maintenance invalidation twice before the DSB solves the problem. If multiple TLB or ICache maintenance invalidations are executed back-to-back, only the last maintenance operation before the DSB needs to be replicated.

or

- Use IPI for TLB and ICache maintenance operations.

There are no errata in this category

2.6. Category B (Rare)

826369: CPU might execute incorrect instructions following a TLBI+DSB sequence

Category B Rare

Products Affected: Cortex-A17 MPCore.

Present in: r0p0

Description

When executing the sequence TLBI followed by a DVM SYNC, ARMv7 architecture expects that all external memory requests using previous TLB mapping are finished on the DVM SYNC. This means that any modification on data on pages that are no-longer executable are not visible by any CPU.

It is possible for Cortex-A12 and Cortex-A17 to start, or use the result of, instruction fetches that used an address targeted by the TLBI operation even after the subsequent DSB/DVM Sync completes.

Configurations affected

This erratum affects all configurations of Cortex-A12 and Cortex-A17.

Conditions

The processor must be executing a code sequence having the following characteristics:

- The CPU receives a TLBI request.

- The CPU receives a DVM SYNC request.

- A CPU changes data on a PA that was mapped before the TLBI, but is no longer mapped as executable.

- The Iside receives data from a linefill for this PA, with the new data. The request for this linefill could have been sent at any previous time, independently of TLBI and DVM SYNC requests.

- The Iside sends the incorrect data to the Core.

Implications

The affected CPU might execute incorrect instructions.

Workaround

The software workaround is to force all CPUs to execute a ISB by broadcasting a software synchronisation request.

827821: CPU might execute stale instructions following broadcast ICI operations

Category B Rare

Products Affected: Cortex-A17 MPCore.

Present in: r0p0

Description

A code-modifying sequence might fail to synchronize instruction fetches of other CPUs with store instructions of the CPU that executes the code-modifying sequence, leading to execution of stale instructions.

Configurations affected

This erratum affects configurations of Cortex-A17 with more than one CPU, or that connects to a coherent interconnect.

Conditions

The processor must be executing a code sequence having the following characteristics:

- CPU A starts a linefill caused by an instruction fetch at address C.

- CPU B executes a code-modifying sequence targeting C (Store, Data Cache Clean, Instruction Cache and Branch Prediction invalidation).

- CPU A and CPU B synchronize to ensure the newly-installed code is visible to CPU A.

- CPU A starts executing the newly-installed code.

The linefill on CPU A completes and returns stale data from memory.

In addition, the erratum only occurs if the Instruction Cache Invalidation instruction described above is either:

- an ICIALL operation.

- an ICIMVA operation broadcast from a CPU in another cluster.

Implications

The affected CPU might execute stale instructions.

Workaround

The software workaround is to add an additional ICIMVA operation at the end of each sequence of ICI operations affected by this erratum (as described in the `Conditions` section), before signalling the completion of that sequence to other CPUs. This additional ICIMVA operation can target any MVA for which address translation will not generate a fault. This includes repeating the last ICIMVA operation.

828419: CPU might execute an Instruction Cache Maintenance Operation by MVA with an incorrect NS descriptor

Category B Rare

Products Affected: Cortex-A17 MPCore.

Present in: r0p0

Description

Under rare timing conditions, a local Instruction cache maintenance operation by MVA might target the Secure alias instead of the Non-secure alias.

Configurations affected

This erratum affects configurations of Cortex-A17 with more than one CPU, or that are connected to a coherent interconnect.

Conditions

The processor must be executing a code sequence having the following characteristics:

- CPU A starts an ICIMVA operation that targets a Non-secure line and misses both in the L1 and in the L2 TLB and therefore launches a page table walk.

- CPU A receives a broadcasted TLB invalidate operation from another CPU or another cluster.

- CPU A then receives a broadcasted Instruction cache maintenance operation from another CPU or another cluster.

- CPU A completes the translation requested by the ICIMVA but cannot directly send the request to the instruction cache because of the previous broadcasted Instruction cache maintenance.

Implications

The affected CPU might not correctly invalidate stale instructions.

Workaround

A potential workaround is to replace ICIMVAU operations by ICIALLUIS operations.

828420: In a multi-cluster system, Cache Clean by MVA to PoC might not be correctly executed

Category B Rare

Products Affected: Cortex-A17 MPCore.

Present in: r0p0

Description

If a line is present in the dirty state in another cluster, it is possible that a Data Clean operation by MVA is not correctly performed and that the line remains in the dirty state in the multi-cluster system and does not become visible externally to a non-coherent agent.

Configurations affected

This erratum affects all configurations of Cortex-A17 in a multi-cluster system.

Conditions

The multi-cluster system must be executing a code sequence with the following characteristics:

- The line at address A is present in cluster A in the dirty state.

- The line at address A is not present in cluster B.

- One of the CPUs in cluster B executes a Data Cache Clean operation to PoC for the line at address A.

- One of the CPUs in cluster B, or the ACP of cluster B, concurrently requests the same line on a data access that will be sent as a ReadShared of a ReadUnique on the AR channel. This can be because of a Load or a Store operation or to a PLD or prefetch operation targeting the L1.

Implications

The affected cluster might not correctly execute the maintenance operation and might keep the affected line in a dirty state without updating the main memory. Therefore, an agent using a non-coherent cacheable buffer that relies on correct execution of the cache maintenance operation might not see the updated value.

Workaround

The software workaround is to treat every Data Clean by MVA to PoC as Data Clean Invalidate by MVA to PoC.

This can be done by setting bit[30] of the CPU diagnostic register (accessible in the cp15 register space at address cp15,0,c15,c0,1). It has the effect of transforming every Data Clean operation in Data Clean Invalidate operation.

This can also be done by replacing the DCCMVAC instruction by a DCCIMVAC in the case of communication between a device and the CPU using a non-coherent cacheable buffer.

832071: Read after read issue with Streaming store and concomitant store by another CPU

Category B Rare

Products Affected: Cortex-A17 MPCore.

Present in: r0p0

Description

Within rare timing constraints, if a CPU is performing a streaming store (write full cache line) and another CPU is performing a store at the same address, a younger load might read the older value of the data (written by the streaming store), while an older load from the same CPU reads the newest value of the data written by the other CPU.

Configurations affected

This erratum affects all configurations of Cortex-A17 and Cortex-A12 with more than one CPU.

Conditions

The following conditions have to be met for the erratum to occur:

- Core0 is writing a full cache line at address A.

- Core0 issues two load operations out-of-order at this address.

- Core1 is also writing at this address.

Implications

When executing the sequence of transactions described above and when particular timing conditions occur, the two load operations might not received the data in the correct order as shown by the following table.

Core0	Core1
STR FULL @A, a0	STR @A, a1
LDR @A, a1	
LDR @A, a0	

Workaround

This case can be avoided by:

- Issuing a DMB between the write of the full cache line and the subsequent load operations

or

- Issuing a DMB between the two load operations.

or

- Disabling the Streaming Mode. This can be done by setting bit[1] of the CPU diagnostic register (accessible in the cp15 register space at address cp15,0,c15,c0,1). Note that streaming performance of the processor is impacted, resulting in a potential 1 to 2% performance drop on mobile workloads.

There are no errata in this category

2.7. Category C

824719: Stage 2 fault might not be reported when address translation for a Cache Maintenance Operation uses a NoAccess Domain

Category C

Products Affected: Cortex-A17 MPCore.

Present in: r0p0

Description

A Stage2 fault on the final Stage1 translation for the address of a Cache Maintenance Operation might not be reported if the address translation indicates the CMO accesses a region using a NoAccess domain.

Configurations affected

This erratum affects all Cortex-A12 and all Cortex-A17 configurations.

Conditions

The erratum occurs if both:

- The address translation indicates the Domain for the targeted area is NoAccess
- The address translation generates one of the following faults during the final Stage1 translation:
  - Access Flag Fault
  - Translation Fault
  - External Abort

Implications

The Cache Maintenance Operation will be performed using an arbitraty physical address, and no Stage2 abort will be generated

Note that this erratum does not cause a data corruption, as all straight invalidating CMOs are translated into the corresponding clean and invalidate cache maintenance operation.

Workaround

The workaround consists, in a Virtualized system, in not making use of the Domains in V7VMSA.

831169: ECC error on L2 DATA RAMs might report an erroneous error index

Category C

Products Affected: Cortex-A17 MPCore.

Present in: r0p0

Description

When an ECC error is detected on L2 DATA RAM, the "Error location" field in L2MRERRSR might report an erroneous value.

Configurations affected

Cortex-A17 configured with:

- an L2 cache size smaller than 8MB.
- ECC implemented.

Conditions

The errata is triggered if:

- an ECC error is detected on the L2 DATA RAM read access.
- the aforementioned access has been triggered by an incoming snoop request from the ACE port.

Implications

The "Error location" field in the L2MRERRSR reports bits[18:6] of the address used for the lookup, independently of the L2 cache size.

Also, when new ECC errors are detected, the "Other error count" and "Same error count" fields might be corrupted.

Workaround

When reading the "Error location" field in L2MRERRSR the value can be masked in software to obtain the real DATARAM index.

There are no errata in this category